PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : **G06T 9/00**	**A1**	**(11) International Publication Number:** **WO 98/29834** **(43) International Publication Date:** 9 July 1998 (09.07.98)

(21) International Application Number: PCT/JP97/04814

(22) International Filing Date: 25 December 1997 (25.12.97)

(30) Priority Data:
60/034,558 30 December 1996 (30.12.96) US

(71) Applicant: SHARP KABUSHIKI KAISHA [JP/JP]; 22–22, Nagaike–cho, Abeno–ku, Osaka–shi, Osaka 545 (JP).

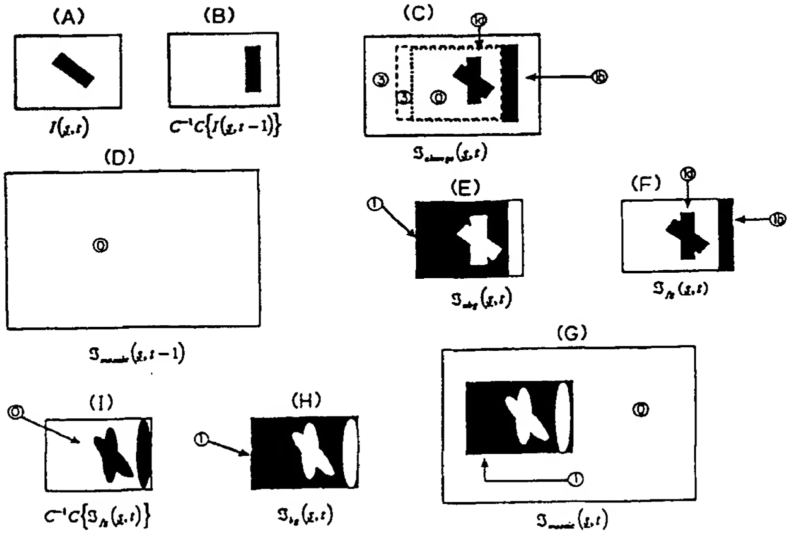
(72) Inventors: CRINON, Regis, Jean, Albert; 2346 N.W. Cascade Street, Camas, WA 98607 (US). SEZAN, Muhammed, Ibrahim; 2213 N.W. Hood Drive, Camas, WA 98607 (US).

(74) Agents: TAKANO, Akichika et al.; Salute Building, 9th floor, 72, Yoshida–cho, Naka–ku, Yokohama–shi, Kanagawa 231 (JP).

(81) Designated States: CN, JP, KR, SG, European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.

(54) Title: SPRITE–BASED VIDEO CODING SYSTEM



(57) Abstract

A sprite–based coding system includes an encoder and decoder where sprite–building is automatic and segmentation of the sprite object is automatic and integrated into the sprite building as well as the coding process. The sprite object is distinguished from the rest of the video objects on basis of its motion. The sprite object moves according to the dominant component of the scene motion, which is usually due to camera motion or zoom. Hence, the sprite–based coding system utilizes dominant motion, to distinguish background images from foreground images. The sprite–based coding system is easily integrated into a video object–based coding framework such as MPEG–4 where shape and texture of individual video objects are coded separately. The automatic segmentation integrated in the sprite–based coding system identifies the shape and texture of the sprite object.

EL327348292US

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania
AM	Armenia
AT	Austria
AU	Australia
AZ	Azerbaijan
BA	Bosnia and Herzegovina
BB	Barbados
BE	Belgium
BF	Burkina Faso
BG	Bulgaria
BJ	Benin
BR	Brazil
BY	Belarus
CA	Canada
CF	Central African Republic
CG	Congo
CH	Switzerland
CI	Côte d'Ivoire
CM	Cameroon
CN	China
CU	Cuba
CZ	Czech Republic
DE	Germany
DK	Denmark
EE	Estonia
ES	Spain
FI	Finland
FR	France
GA	Gabon
GB	United Kingdom
GE	Georgia
GH	Ghana
GN	Guinea
GR	Greece
HU	Hungary
IE	Ireland
IL	Israel
IS	Iceland
IT	Italy
JP	Japan
KE	Kenya
KG	Kyrgyzstan
KP	Democratic People's Republic of Korea
KR	Republic of Korea
KZ	Kazakstan
LC	Saint Lucia
LI	Liechtenstein
LK	Sri Lanka
LR	Liberia
LS	Lesotho
LT	Lithuania
LU	Luxembourg
LV	Latvia
MC	Monaco
MD	Republic of Moldova
MG	Madagascar
MK	The former Yugoslav Republic of Macedonia
ML	Mali
MN	Mongolia
MR	Mauritania
MW	Malawi
MX	Mexico
NE	Niger
NL	Netherlands
NO	Norway
NZ	New Zealand
PL	Poland
PT	Portugal
RO	Romania
RU	Russian Federation
SD	Sudan
SE	Sweden
SG	Singapore
SI	Slovenia
SK	Slovakia
SN	Senegal
SZ	Swaziland
TD	Chad
TG	Togo
TJ	Tajikistan
TM	Turkmenistan
TR	Turkey
TT	Trinidad and Tobago
UA	Ukraine
UG	Uganda
US	United States of America
UZ	Uzbekistan
VN	Viet Nam
YU	Yugoslavia
ZW	Zimbabwe

DESCRIPTION

SPRITE-BASED VIDEO CODING SYSTEM

Field of the Invention

This invention relates to a mechanism by which a sprite (also called mosaic) is built automatically both in an encoder and a decoder, operating in a separate shape/texture coding environment such as MPEG-4. We also discuss applications that will utilize this technology.

Background of the Invention

A mosaic image (the terms mosaic and sprite will be used interchangeably) is built from images of a certain scene object over several video frames. For instance, a mosaic of the background scene in case of a panning camera will result in a panoramic image of the background.

In MPEG-4 standardization activities, two major types of sprites and sprite-based coding are defined. The first type is called *off-line static sprite*. An off-line static sprite is a panoramic image which is used to produce a sequence of snapshots of the same video object (such as background). Each individual snapshot is generated by simply warping portions of the mosaic content and copying it to the video buffer where the current video frame is being reconstructed. Static sprites are built off-line and are transmitted as side information.

The second type of mosaic is called *on-line dynamic sprite*. On-line dynamic sprites are used in predictive coding of a video object. A prediction of each snapshot of the video object in a sequence is obtained by warping a section of the dynamic sprite. The residual signal is coded and used to update the mosaic in the encoder and the decoder concurrently. The content of a dynamic mosaic may be constantly updated to include the latest video object information. As opposed to static sprites, dynamic sprites are built on line simultaneously in the encoder and decoder. Consequently, no additional information needs to be transmitted.

Summary of the Invention

We have described a syntax for MPEG-4 which provided a unified syntax [2] for off-line static sprite and on-line dynamic sprite-based coding. Our syntax also allows new modes that we refer as "dynamic off-line sprite-based coding," where predictive coding is performed on the basis of an off-line sprite (as opposed to directly copying the warped sprite as in the case of off-line static sprites), and "on-line static sprite-based coding," where the encoder and the decoder stop building the sprite further, and use it as a static sprite whether it is partially or fully completed.

Both off-line static and on-line dynamic sprite-based coding require constructing a sprite. In the former case, the sprite is built prior to transmission. In the later case, the sprite is built on-line during the transmission. So far, MPEG-4 has assumed that the outline (segmentation) of the object for which the sprite is going to be built is known *a-priori* at every time instant. Although this is true in certain applications, especially in post-production or content generation using blue screen techniques, automatic segmentation should be an integral part of sprite building in general. There is therefore a need for sprite-based coding systems where sprite building does not require *a-priori* knowledge of scene segmentation.

In this disclosure, we describe a sprite-based coding system (encoder and decoder) where sprite-building is automatic and segmentation of the sprite object is automatic and integrated into the sprite building as well as the coding process.

We assume that the sprite object can be distinguished from the rest of the video objects on basis of its motion. We assume that the sprite object moves according to the dominant component of the scene motion, which is usually due to camera motion or zoom. Hence, our system utilizes dominant motion, which is known to those of skill in the art.

Our system is suitable for a video object-based coding framework such as MPEG-4 [3], where shape and texture of individual video objects are coded separately. The automatic segmentation integrated in the described system identifies the shape and texture of the sprite object.

There are several possible applications of the invention: In very low bit rate applications, coding of video frames in terms of video objects within may be expensive, because the shape of such objects may consume a significant portion of the limited bit budget. In such cases, our system can fallback to frame-based coding where automatic segmentation is only used to obtain better dominant motion estimation for sprite building and dominant motion-compensated prediction, as described in the "Operations" section, later herein.

The described coding system has features which make it suitable for applications where camera view may change frequently, such as video conferencing with multiple cameras, or a talk show captured with more than one camera. Our system may be applied to building multiple sprites and using them as needed. For instance, if the camera goes back and forth between two participants in front of two different backgrounds, two background sprites are built and used as appropriate. More specifically, when back ground A is visible, building of the sprite for background B and its use in coding is suspended until Background B appears again. The use of multiple sprites in this fashion is possible within the MPEG-4 framework, as will be described in the "Operations" section.

The disclosed system generates a sprite during the encoding process as will be described later herein. However, the resulting sprite may be subsequently used, after coding, as a representative image of the compressed video clip. Its features can be used to identify the features of the video clip which can then be used in feature-based (or content-based) storage and retrieval of video clip. Hence sprite-based coding provides a natural fit to populating a video library of bitstreams where sprite images generated during the encoding process act as representative images of the video clips. Indeed the mosaics can also be coded using a still image coding method. Such a video library system is depicted in Fig. 5.

In a similar fashion, one or several event lists may be associated with a background sprite. A possible choice for an event list is the set of consecutive positions of one or several vertices belonging to each foreground objects. Such a list can then be used to generate token representative image of the foreground object position in the sprite. Consecutive positions of each vertex could either be linked by a straight line or could share a distinct color. The consecutive positions of the vertex may be shown statically (all successive positions in the same sprite) or dynamically (vertex positions shown in the mosaic successively in time). A vertex here can be chosen to correspond to any distinctive feature of the foreground object, such as the center of gravity or a salient point in the shape of the object. In the latter case, and if several vertices are used simultaneously, the vertices might be arranged according to a hierarchical description of the object shape. With this technique, a user or a presentation interface has the freedom to chose between coarse to finer shapes to show successive foreground object positions in the background sprite. This concept may be used in a video library system to retrieve content based on motion characteristics of the foreground.

The automatic sprite building portion of the described system may be used in an off-line mode in a video conferencing application where the off-line sprite is built prior to transmission. Depiction of such a system is shown in Fig. 6. The described system can also generate a sprite that has a higher spatial resolution than the original images.

Brief Descriptions of the Drawings

Fig. 1 illustrates the steps used in the method of the invention at time t-1.

Fig. 2 illustrates the steps used in the method of the invention at time t to t+1.

Fig. 3 illustrates the steps used in the method of the invention at time t+1 to t+2.

Fig. 4 is a block diagram of the method of the invention.

Fig. 5 is a block diagram of the system of the invention.

Fig. 6 depicts the system and method of the invention as used in a video conferencing system.

Fig. 7 depicts how consecutive portions of a foreground object may be represented in a mosaic according to the invention.

Detailed Description of the Preferred Embodiments

The described method is designed to progressively learn to dissociate foreground from background while building a background mosaic at the same time. Steps 1 to 10 are repeated until the construction of the background is complete or until it is aborted.

Assumptions

The notations are as follows:

$I(\underline{s},t)$ denotes the content of video frame at spatial position $\underline{s}$ and at time t .

$W_{t\leftarrow(t-1)}\left(I(\underline{s},t-1)\right)$ denotes a warping operator which maps the image at time (t-1) to time t. For a given pixel location $\underline{s}_0$ in a video buffer at time t, this warping operation is performed by copying the pixel value at corresponding location $\underline{s}$ in frame (t-1).The correspondence between location $\underline{s}_0$ and location $\underline{s}$ is established by a particular and well defined transformation such as an affine or a perspective transformation.

$\Im(\underline{s},t)$ is an indicator buffer, say for quantity x, which can be either 1 or 2 bits deep for all spatial location $\underline{s}$.

Thresh is a threshold value. The operations $\leq Thresh$ and $> Thresh$ are symbolic and can represent complex thresholding operations.

The size (per color component) of the current image frame $I(\underline{s},t)$ is $M_t \times N_t$ and the size of the previous compressed/decompressed frame after warping, $W_{t\leftarrow(t-1)}\left(C^{-1}C\left\{I(\underline{s},t-1)\right\}\right)$, is such that it can be inscribed in a rectangular array of $M_{t-1} \times N_{t-1}$ pixels.

The sprite $M(\underline{s},t)$ is an image intensity (texture) buffer of size $M_m \times N_m$ per color component. The field $\Im_{mosaic}(\underline{s},t)$ is a single component field of the same size.

The construction of the sprite is started at time t. The image $I(\underline{s},t-1)$ has already been compressed and decompressed and it is available in both the encoder and the decoder.

In the following steps, the image content is assumed to have a background and a foreground part (or VO) and a mosaic of the background is built

Step 1: Initialization.

Referring now to Figs. 1-3, the results of the steps of the method described in the previous section are depicted. Fig. 1 illustrates steps 0 through 11 from time t-1, the instant when mosaic building is initiated, to time t when the a new video frame or field has been acquired. Figs. 2 and 3 illustrate steps 2 through 11 from time t to t+1 and from time t+1 to t+2, respectively. At the top left corner in each of these figures (A) is shown the newly acquired video frame which is compared to the previous video frame (next image field to the right)(B) once it has been compressed/de-compressed and warped (step 2). Step 3 is illustrated by the rightmost image field (C) in the first row of each figure. This field shows the area where content change has been detected. The status of the mosaic buffer is shown in the leftmost image field in the second row (D). This buffer is used to identify the new background areas as described in step 4. These areas correspond to regions where background was not known until now. Foreground identification is illustrated by the rightmost image in the second row (F). The operations associated with this image are described in step 5 which use the change map, the mosaic and the new background areas to define the foreground. Steps 6 and 7 of the method are illustrated by the two leftmost image fields in the third row (G, H). Here, background information comes from the compressed/decompressed foreground information obtained in the previous step. Finally , the mosaic updating process is illustrated by the bottom right image field (I). This process takes place in steps 8,9,10 and 11 of the method.

The binary field $\Im_{mosaic}(\underline{s},t)$ is initialized to 0 for every position $\underline{s}$ in the buffer, meaning that the content of the mosaic is unknown at these locations.

The content of the mosaic buffer $M(\underline{s},t)$ is initialized to 0.

The warping parameters from the current video frame $I(\underline{s},t-1)$ to the mosaic is initialized to be $W_{t0\leftarrow(t-1)}(\ )$, t0 here representing an arbitrary fictive time. This initial warping is important as it provides a way to specify the "resolution" or the "time reference" used to build the mosaic. Potential applications of this initial mapping are making a mosaic with super spatial resolution or selection of an optimal time t0 minimizing the distortion introduced by the method. These initial warping parameters are transmitted to the decoder.

Step 2: Acquisition.

The image $I(\underline{s},t)$ is acquired and the forward warping parameters for mapping the image $I(\underline{s},t-1)$ to $I(\underline{s},t)$are computed. The number of warping parameters as well as the method for estimating these parameters are not specified here. A dominant motion estimation algorithm such as the one given in [4] may be used. The warping parameters are composed with the current warping parameters, resulting in the mapping $W_{t\leftarrow t0}(\ )$. These parameters are transmitted to the decoder.

Step 3: Detect Change in Content Between Previously Coded/Decoded Frame and Current Frame.

i) Initialization of a large buffer of size $M_b \times N_b$ greater than the image ($M_b > M_t$, $N_b > N_t$) and possibly as large as the mosaic. The buffer is 2 bits deep at every location. The buffer is initialized to 3 to indicate unknown status.

$$\Im_{change}(\underline{s},t) = 3$$

ii) Compute (motion compensated) scene changes over common image support. Give label 0 to all locations where change in content is deemed small. Give label 1a to locations where change is detected to be large. To make regions more homogeneous, implement additional operations (e.g. morphological operations) which either reset label from 1a to 0 or set label from 0 to 1a. Regions labeled 0 will typically be considered and coded as part of the background Video Object while regions labeled 1a will typically be encoded as part of the foreground Video Object.

$$\Im_{change}(\underline{s},t) = \begin{cases} 0 \quad if \quad \left| I(\underline{s},t) - W_{t\leftarrow t-1}\left(C^{-1}C\left\{I(\underline{s},t-1)\right\}\right)\right| \le Thres_{change} \\ 1a \quad otherwise \end{cases}$$

where $Thres_{change}$ denotes a pre-defined threshold value.

iii) Tag new image regions, where support of image at time t does not overlap with support of image at time (t-1), as

$$\Im_{change}(\underline{s},t) = 1b$$

Step 4: Identify New Background Areas.

A new background area is detected if there has not been any change in image content in the last two video frames. The corresponding area in the mosaic must also indicate that the background at this location is unknown. The resulting new background area is then pasted to any neighboring regions where background is known. As will be seen in later steps, incorporation of new background data into the mosaic must be done according to compressed/de-compressed background shape information to avoid any drift between encoder and decoder.

$$\Im_{nbg}\left(\underline{s},t\right)=\begin{cases}1 \ \ if \ \ \left(\left(\Im_{change}\left(\underline{s},t\right)==0\right) \ \&\& \ \left(W_{t\leftarrow t0}\left(\Im_{mosaic}\left(\underline{s},t-1\right)\right)==0\right)\right) \\ 0 \ \ otherwise\end{cases}$$

Here, the indicator value 0 means that the background is unknown.

Step 5: Perform Foreground/Background Segmentation.

First look at regions where the background is known ($\Im_{mosaic}\left(\underline{s},t-1\right)=1$). Perform thresholding to distinguish the foreground from the background (case (i)). For regions where background is not known, tag as foreground any regions where changes have occurred (label 1a and 1b defined in step 3) (cases (iii) and (iv)).

Case (ii) represents new background areas which are excluded from being part of the foreground.

i) If $W_{t\leftarrow t0}\left(\Im_{mosaic}\left(\underline{s},t-1\right)\right)==1$

$$\Im_{fg}(\underline{s},t)=\begin{cases}=1a \ \ if \ \ \left| \ I(\underline{s},t) - W_{t\leftarrow t0}(\ M(\underline{s},t-1)) \ \right| > Thresh_{fg} \\ =0 \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad otherwise\end{cases}$$

where $Thres_{fg}$ is a pre-defined threshold value which is used here to segment foreground from background.

ii) else if $\Im_{nbg}\left(\underline{s},t\right)==1$

$$\Im_{fg}\left(\underline{s},t\right)=0$$

iii) else if $\left(\left(\Im_{mosaic}\left(\underline{s},t-1\right)==0\right) \ \&\& \ \left(\Im_{change}\left(\underline{s},t\right)==1a\right)\right)$

$$\Im_{fg}\left(\underline{s},t\right)=1a$$

iv) else $\left(\left(\Im_{mosaic}\left(\underline{s},t-1\right)==0\right) \ \&\& \ \left(\Im_{change}\left(\underline{s},t\right)==1b\right)\right)$

$$\Im_{fg}\left(\underline{s},t\right)=1b$$

In cases (iii) and (iv), a sub-classification of the regions tagged as 1 into either regions 1a and 1b is used for the sole purpose of providing the encoder with the flexibility to follow different macroblock selection rules. For example, regions tagged as 1a might be preferably coded with inter-frame macroblocks since these regions occur over common image support. On the other hand, regions tagged as 1b might preferably be coded with intra-frame macroblocks since these regions do not share a common support with the previous frame.

Step 6: Compress/Decompress Foreground Shape and Texture.

Use conventional (I, P or B-VOP) prediction mode to encode foreground regions labeled as 1a and 1b. In the case of P or B-VOPs, individual macroblocks can either use inter-frame prediction or intra-frame coding. The pixels corresponding to regions labeled 1b (newly revealed background not represented in mosaic) are favored to be coded as intra macroblocks. The shape of the foreground is compressed and transmitted as well. Once de-compressed, this shape is used by the encoder and decoder to update the content of the mosaic. This process can be performed using the MPEG-4 VM 5.0 [3].

Step 7: Get Background Shape.

Get background shape from compressed/de-compressed foreground shape. Compression/De-compression is necessary here to ensure that encoder and decoder share the same shape information.

$$\Im_{bg}(\underline{s},t) = \begin{cases} 1 \;\; if \;\; C^{-1}C\{\Im_{fg}(\underline{s},t) == 0\} \\ 0 \;\; otherwise \end{cases}$$

where $C^{-1}C\{\ \}$denotes shape coding/decoding which for instance can be performed as described in [3].

Step 8: Initialize New Background Texture in Mosaic.

Identify regions where new background has occurred and initialize mosaic with content found in previous video frame (time (t-1)). Note that the field $\Im_{nbg}(\underline{s},t)$ cannot be used here since this information is unknown to the decoder.

$$M'(\underline{s},t-1) = \begin{cases} M(\underline{s},t-1) \quad if \;\; (\Im_{mosaic}(\underline{s},t-1) == 1) \\ W_{t0\leftarrow(t-1)}\left(C^{-1}C\{I(\underline{s},t-1)\}\right) \;\; if \;\; \left(\left(W_{t0\leftarrow t}(\Im_{bg}(\underline{s},t)\right) == 1\right) \;\&\&\; \left(\Im_{mosaic}(\underline{s},t-1) == 0\right)\right) \end{cases}$$

Step 9: Calculate Background Texture Residuals From Mosaic Prediction.

If $\Im_{bg}(\underline{s},t) == 1$, calculate difference signal by using mosaic content as predictor: The resulting $\Delta I(\underline{s},t)$ is used to compute the difference signal over the entire macroblock where the pixel $(\underline{s},t)$ is located. This difference signal is compared to conventional difference signals produced by using prediction from the previous and the next video frame (P or B prediction mode). The macroblock type is selected according to the best prediction mode. The residual signal is transmitted to the decoder along with the compressed background shape as described in [2].

$$\Delta I(\underline{s},t) = I(\underline{s},t) - W_{t\leftarrow t0}\left(M'(\underline{s},t-1)\right)$$

Step 10: .Update Background Shape in Mosaic.

Update mosaic map to include shape of new background.

$$\Im_{mosaic}(\underline{s},t) = \begin{cases} 1 \;\; \mathit{if} \;\; \Im_{mosaic}(\underline{s},t-1) == 1 \\ 1 \;\; \mathit{if} \;\; \left(\left(W_{t0\leftarrow t}\left(\Im_{bg}(\underline{s},t)\right) == 1\right) \;\&\&\; \left(\Im_{mosaic}(\underline{s},t-1) == 0\right)\right) \\ 0 \;\; \mathit{otherwise} \end{cases}$$

Step 11: Update Mosaic.

Update content of the mosaic in regions corresponding to new or non-covered background in frame t

$$M(\underline{s},t) = \left[1 - \alpha W_{t0\leftarrow t}\left(\Im_{bg}(\underline{s},t)\right)\right] M'(\underline{s},t-1) +$$
$$\alpha W_{t0\leftarrow t}\left(\Im_{bg}(\underline{s},t)\right)\left[M'(\underline{s},t-1) + W_{t0\leftarrow t}\left(C^{-1}C\{\Delta I(\underline{s},t)\}\right)\right]$$

The selection of the value of the blending parameter α $(0 < \alpha < 1)$ in the above equation is application dependent.

The method described above builds the mosaic with reference to time *t0,* which can be a time instant in the past or can be the current time or a future time instant. It is straightforward to rewrite the above equations for the case where the mosaic is continuously warped to the current time instant t.

Turning now to Fig. 4, a block diagram of the method is depicted. The purpose of this drawing is to highlight the dependencies in the various components and quantities used by the method of the invention. It also emphasizes the various warping and un-warping stages that are necessary to align consecutive video fields.

Fig. 5 shows a block diagram of the digital video database system that uses the method of the current invention.

Fig. 6 shows a block diagram of a video conferencing system that use an off-line built background sprite as dynamic sprite during transmission

Fig. 7 shows an example of how consecutive positions of a foreground object (here a car) may be represented in a mosaic by plotting the successive positions of one or several salient points (V) belonging to the shape of the foreground. The color of the vertices is changed from t0 to t0+1 and from t0+1 to t0+2 to avoid any confusion. In this example, the vertices are shown statically in the mosaic and they capture one level of shape description only.

Operation of the Various Embodiments

A Mosaic-Based Video Conferencing and Videophone System.

Referring now to Figs. 5 and 6, the communication protocol can include a configuration phase (time adjustable) during which an on-line background mosaic is being built. During this period, each videophone uses the small displacements of the head and shoulder to build a background mosaic. The displacements of the foreground can be voluntary (system guides user) or not (no gain in coding efficiency if foreground does not move). In this case the method described above is used to build the background mosaic. During normal video transmission, the mosaic is used as a dynamic sprite and the blending factor is set to 0 to prevent any updating. In this case, macroblock types may be dynamic or static. In one extreme case, all macroblocks are static-type macroblocks meaning that the background mosaic is being used as a static sprite. In another extreme case, all macroblocks are of type dynamic and the mosaic is being used as a dynamic (predictive) sprite. This later case requires a higher data transmission bandwidth. Alternatively, a mosaic of the background scene can be built before the transmission and then be used as a static or a dynamic sprite during normal transmission session.

A Mosaic-Based Video Database

The above method may be used in populating and searching a database of video bitstreams, i.e., a database of compressed bitstreams. In such a system, video clips are compressed using the above method. The result is a compressed bitstream and a mosaic generated during the encoding process. The mosaic image can be used as a representative image of the video clip bitstream and its features can be used in indexing and retrieval of the bitstream belonging to that video clip.

Furthermore, motion trajectory of the foreground can be overlaid on top of mosaic to provide user with rough description of foreground motion in the sequence. Trajectory of a foreground object can be represented by a set of points, each representing the position of a particular feature of the foreground object at a given instant. The feature points can be salient vertices of the object shape. A hierarchical description of object shape would bring the additional advantage of allowing the database interface to overlay from coarse to fine shape outlines in the mosaic. Consecutive vertex positions can be shown together in the same background mosaic or could be displayed successively in time with the same mosaic support. Note that this idea provides the additional benefit of facilitating motion-based retrieval since the motion of the foreground is represented in mosaic reference space.

Referring now to Fig. 7, the background mosaic is comprised of the grass, sky, sun and tree. The foreground object is a car subject to an accelerated motion and moving from left to right. The shape of the car is shown in black. Eight vertices "V" have been selected to represent this shape. Fig. 7 shows that consecutive positions of the car can be represented in the mosaic by simply plotting the vertices at their successive positions. The color of the vertices is changed from t0 to t0+1 and from t0+1 to t0+2 to avoid any possible confusion. In this example, the vertices are shown statically in the mosaic and they capture one level of shape description only. Finally, mosaic can be used as an icon. By clicking on the mosaic icon, user would trigger playback of the sequence.

Support of Multiple Mosaics in Applications with Frequent Scene Changes.

In the case where the video sequence includes rapid and frequent changes from one scene to another, as may be the case in video conferencing applications, it is desirable to build two or more (depending on how many independent scenes there are) mosaics simultaneously. Having more than one mosaic does not force the system to re-initiate the building of a new mosaic each time a scene cut occurs. In this framework, a mosaic is used and updated only if the video frames being encoded share similar content. Note that more than one mosaic can be updated at a time since mosaics are allowed to overlap.

Optimal Viewport.

The arbitrary mapping $W_{(t-1)\leftarrow t0}(\ )$ used at the beginning of the method can be used to represent the optimal spatial representation domain for the mosaic where distortion and artifacts are minimized. While this is at this point an open problem that will require further study on our part, there is little doubt that the possibility exists to find an optimal mosaic representation where ambiguities (parallax problems) and/or distortion are minimized according to pre-defined criterion.

Improved Resolution

Likewise, the arbitrary mapping $W_{(t-1)\leftarrow t0}(\ )$can include a zooming factor which has the effect of building a mosaic whose resolution is potentially 2,3, or N times larger than the resolution of the video frames used to build it. The arbitrary fixed zooming factor provides a mechanism by which fractional warping displacements across consecutive video frames are recorded as integer displacements in the mosaic. The larger the zooming factor, the longer the sequence must be before the mosaic can be completed (more pixel locations to fill up). The MPEG-4 framework allows the implementation of such a scheme.

We denote this arbitrary mapping $W_{res}()$. In the linear case, this operator is the identity matrix multiplied by a constant scalar greater than 1. This scaling factor defines the enlargement factor used for the mosaic. The mosaic update equation shown in step 11 can be re-written as follows.

$$M(\underline{s},t) = \left[1 - \alpha\, W_{res}\left(W_{t0\leftarrow t}\left(\Im_{bg}(\underline{s},t)\right)\right)\right] M'(\underline{s},t-1) + \\ \alpha W_{res}\left(W_{t0\leftarrow t}\left(\Im_{bg}(\underline{s},t)\right)\right)\left[M'(\underline{s},t-1) + W_{res}\left(W_{t0\leftarrow t}\left(C^{-1}C\{\Delta I(\underline{s},t)\}\right)\right)\right]$$

This equation shows that the mosaic is being built at the fixed time t0 which can be the time corresponding to the first video frame, the time corresponding to the final frame or any other time in between. In this case, the arbitrary mapping $W_{res}()$, is always composed with the warping transformation $W_{t0\leftarrow t}$. When the mosaic is continuously warped toward the current video frame, the update equation must be re-written as follows:

$$M(\underline{s},t) = \left[1 - \alpha\left(\Im_{bg}(\underline{s},t)\right)\right] W_{t\leftarrow(t-1)}\left(M'(\underline{s},t-1)\right) + \\ \alpha\left(\Im_{bg}(\underline{s},t)\right)\left[W_{t\leftarrow(t-1)}\left(M'(\underline{s},t-1)\right) + W_{res}\left(C^{-1}C\{\Delta I(\underline{s},t)\}\right)\right]$$

The equation above shows that the arbitrary mapping $W_{res}()$ is no longer composed with the frame-to-frame warping operator $W_{t\leftarrow(t-1)}$ but instead applied to the compressed/de-compressed residuals. In MPEG-4, the arbitrary operator $W_{res}()$ can be transmitted with appropriate extension of the syntax, as the first set of warping parameters, which currently supports only positioning the first video frame in the mosaic buffer via a translational shift.

Coding of Video Sequences at Very Low Bit Rates.

- In very low bit rate applications, the transmission of shape information may become an undesirable overhead. The method described above can still operate when transmission of shape information is turned off. This is accomplished by setting background shape to one at every pixel (step 7) and setting the blending factor α to 1 (step 11). The latter setting guarantees that the mosaic will always display the latest video information which is a necessity in this situation since foreground is included in the mosaic. In this situation, the macroblock types can be either intra, inter, static sprite or dynamic sprite. The sprite is being used as a static sprite if all macroblocks are of type static. This is the most likely situation for a very low bit rate application since no residual is transmitted in this case. The sprite is being used as a dynamic sprite if all macroblocks are of type dynamic.

CLAIMS

1. A method of sprite-based predictive video coding (encoding and decoding) where sprite-building is automatic, and segmentation of the sprite object is automatic and integrated into the sprite building as well as the encoding and decoding processes, comprising:

initializing a binary field to zero for every position in a buffer;

acquiring the image and forwarding warping parameters for the image for mapping;

detecting any change in content between a previously coded/decoded frame and the current frame;

identifying new background areas;

segmenting the foreground and background;

preparing foreground shape and texture by compressing or decompressing the subject shapes;

deriving the background shape from the previously prepared foreground shape;

initializing the new background texture in mosaic;

determine the background texture residuals from the mosaic prediction;

updating the background shape mosaic; and

updating the mosaic in all regions corresponding to new or non-covered background.

2. A compressed video database system wherein sprites built during encoding are used as representative images of input video clips that can be analyzed and indexed for storage and retrieval purposes, comprising:

a sprite-based encoder for receiving a video clip and generating a video bitstream and a mosaic;

a feature extractor for extracting features from said mosaic and for identifying representative features;

a video database generator for generating a video database from said representative features and said video bitstream; and

a search engine for searching said video database for selected representative features.

FIG.1

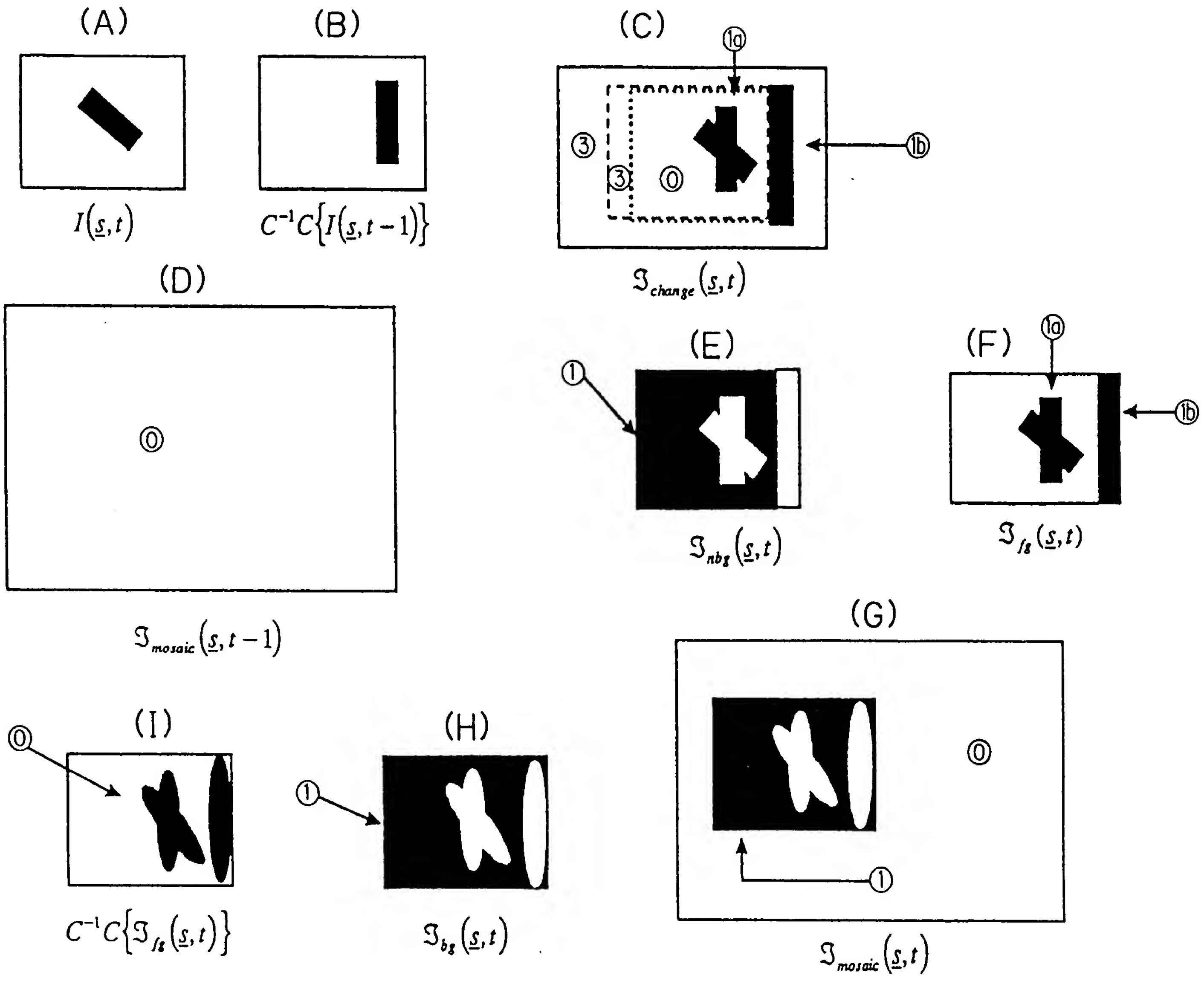
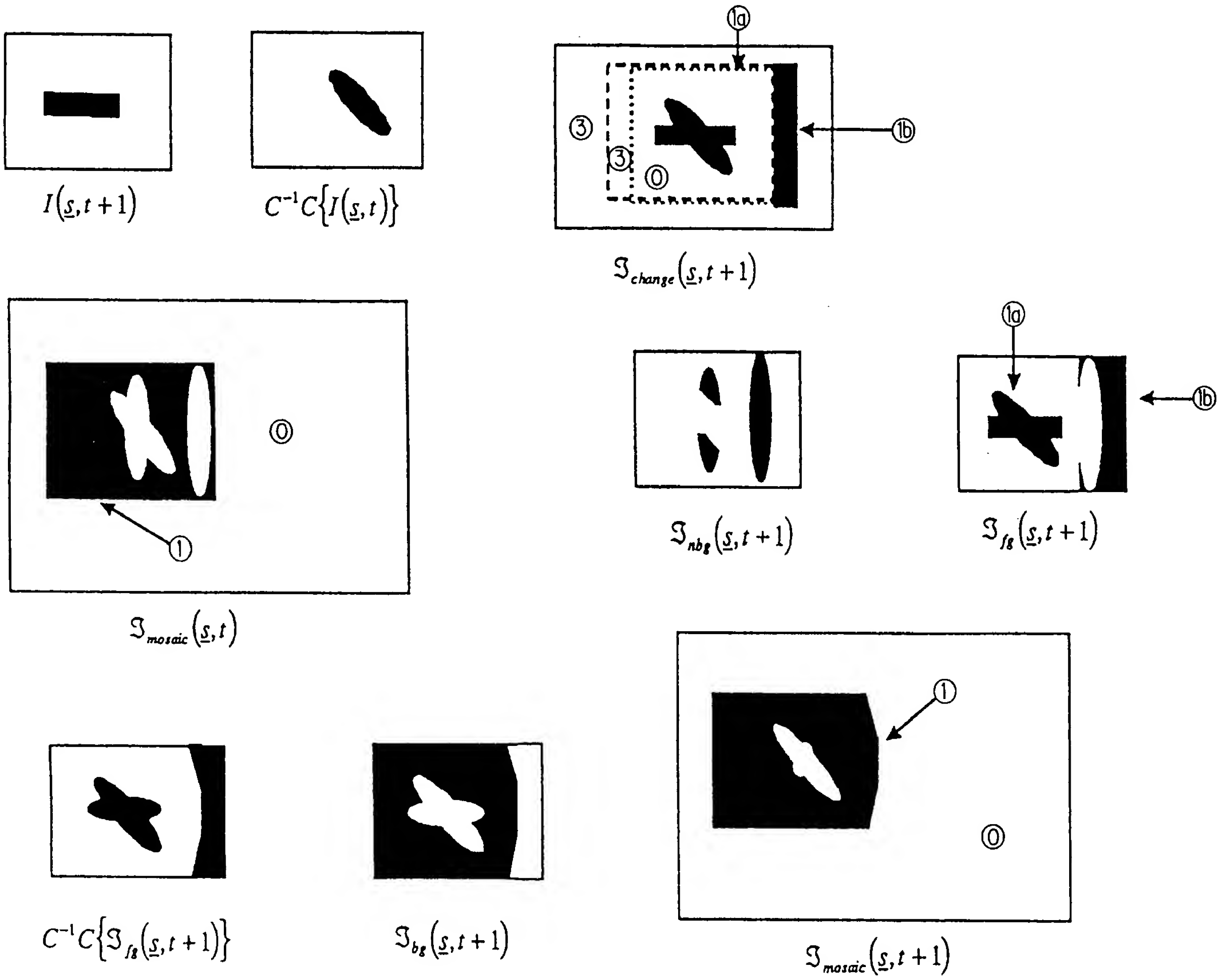

(A)
$I(\underline{s},t)$
(B)
$C^{-1}C\{I(\underline{s},t-1)\}$
(C)
1a
1b
3
3
0
$\Im_{change}(\underline{s},t)$
(D)
0
$\Im_{mosaic}(\underline{s},t-1)$
(E)
1
$\Im_{nbg}(\underline{s},t)$
(F)
1a
1b
$\Im_{fg}(\underline{s},t)$
(G)
0
1
$\Im_{mosaic}(\underline{s},t)$
(I)
0
$C^{-1}C\{\Im_{fg}(\underline{s},t)\}$
(H)
1
$\Im_{bg}(\underline{s},t)$

FIG.2



$I(\underline{s},t+1)$

$C^{-1}C\{I(\underline{s},t)\}$

$\Im_{change}(\underline{s},t+1)$

$\Im_{mosaic}(\underline{s},t)$

$\Im_{nbg}(\underline{s},t+1)$

$\Im_{fg}(\underline{s},t+1)$

$C^{-1}C\{\Im_{fg}(\underline{s},t+1)\}$

$\Im_{bg}(\underline{s},t+1)$

$\Im_{mosaic}(\underline{s},t+1)$

FIG.3

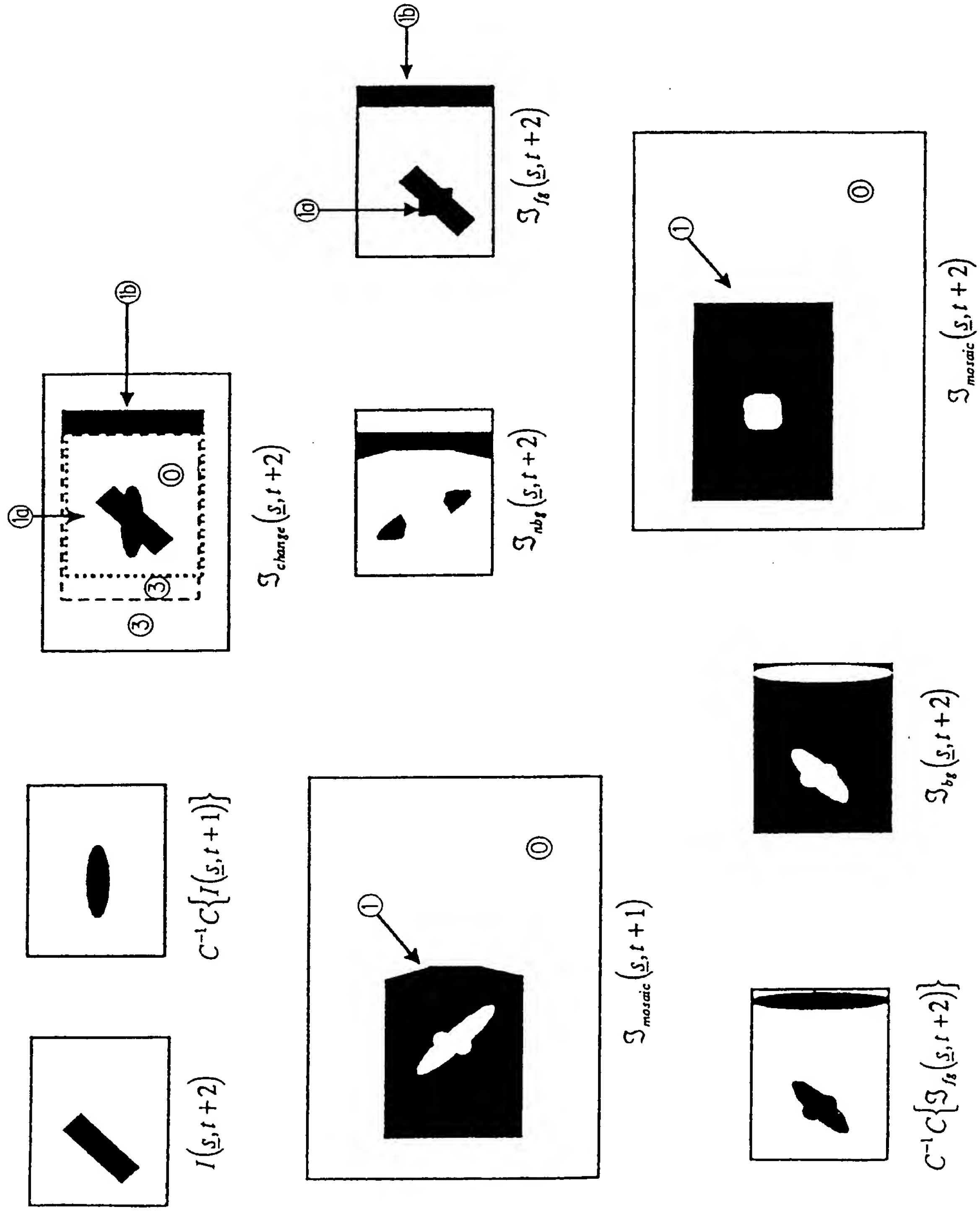

$I(\underline{s},t+2)$
$C^{-1}C\{I(\underline{s},t+1)\}$
$\Im_{change}(\underline{s},t+2)$
1a
1b
0
3
3
$\Im_{mosaic}(\underline{s},t+1)$
1
0
$\Im_{nbg}(\underline{s},t+2)$
$\Im_{fg}(\underline{s},t+2)$
1a
1b
$C^{-1}C\{\Im_{fg}(\underline{s},t+2)\}$
$\Im_{bg}(\underline{s},t+2)$
$\Im_{mosaic}(\underline{s},t+2)$
1
0

FIG.4

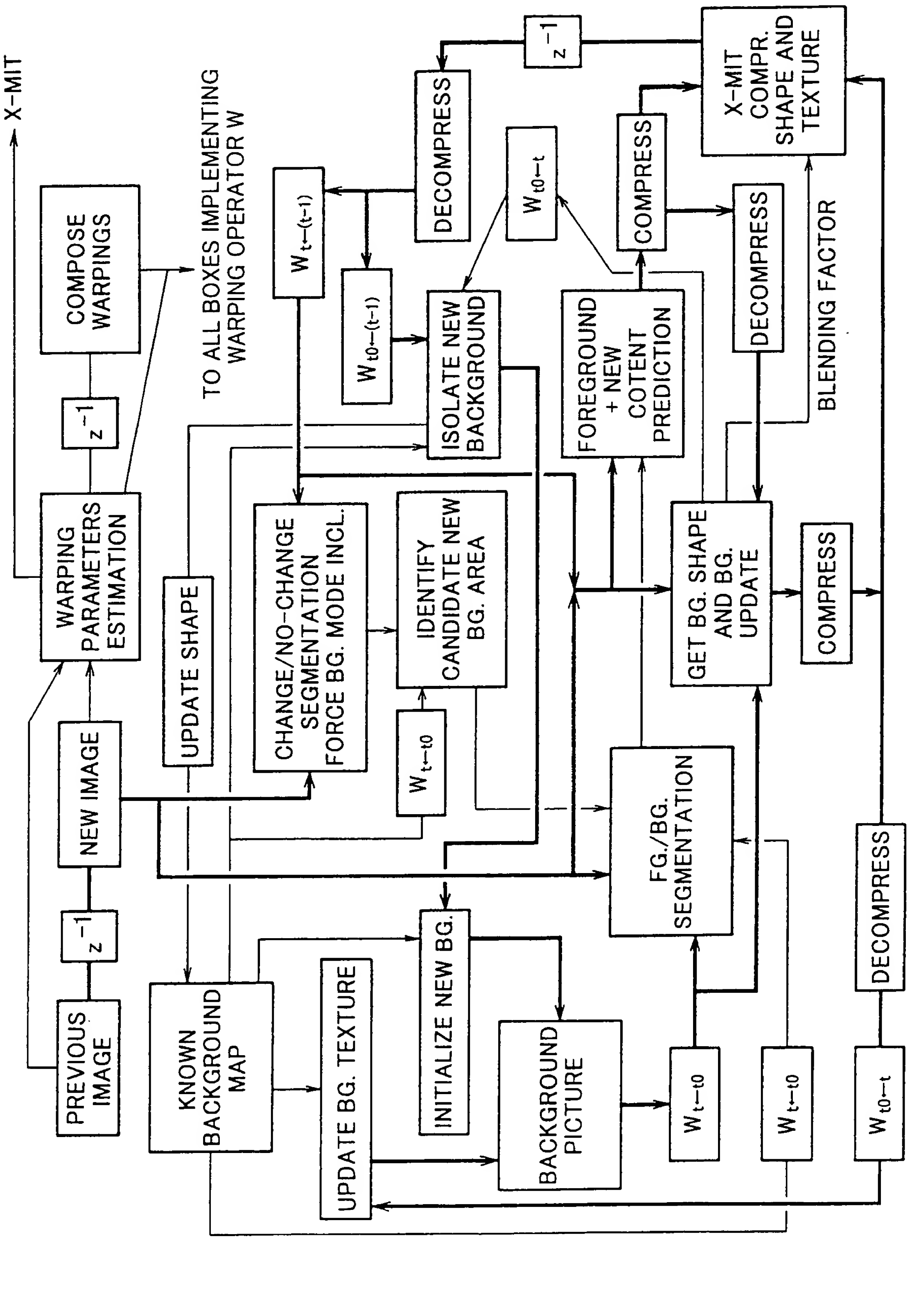

PREVIOUS IMAGE
z^{-1}
NEW IMAGE
WARPING PARAMETERS ESTIMATION
z^{-1}
COMPOSE WARPINGS
X-MIT
TO ALL BOXES IMPLEMENTING WARPING OPERATOR W
KNOWN BACKGROUND MAP
UPDATE SHAPE
CHANGE/NO-CHANGE SEGMENTATION FORCE BG. MODE INCL.
$W_{t\leftarrow(t-1)}$
$W_{t0\leftarrow(t-1)}$
DECOMPRESS
UPDATE BG. TEXTURE
INITIALIZE NEW BG.
$W_{t\leftarrow t0}$
IDENTIFY CANDIDATE NEW BG. AREA
ISOLATE NEW BACKGROUND
$W_{t0\leftarrow t}$
BACKGROUND PICTURE
FG./BG. SEGMENTATION
FOREGROUND + NEW COTENT PREDICTION
COMPRESS
z^{-1}
$W_{t\leftarrow t0}$
GET BG. SHAPE AND BG. UPDATE
DECOMPRESS
X-MIT COMPR. SHAPE AND TEXTURE
$W_{t\leftarrow t0}$
COMPRESS
BLENDING FACTOR
$W_{t0\leftarrow t}$
DECOMPRESS

FIG.5

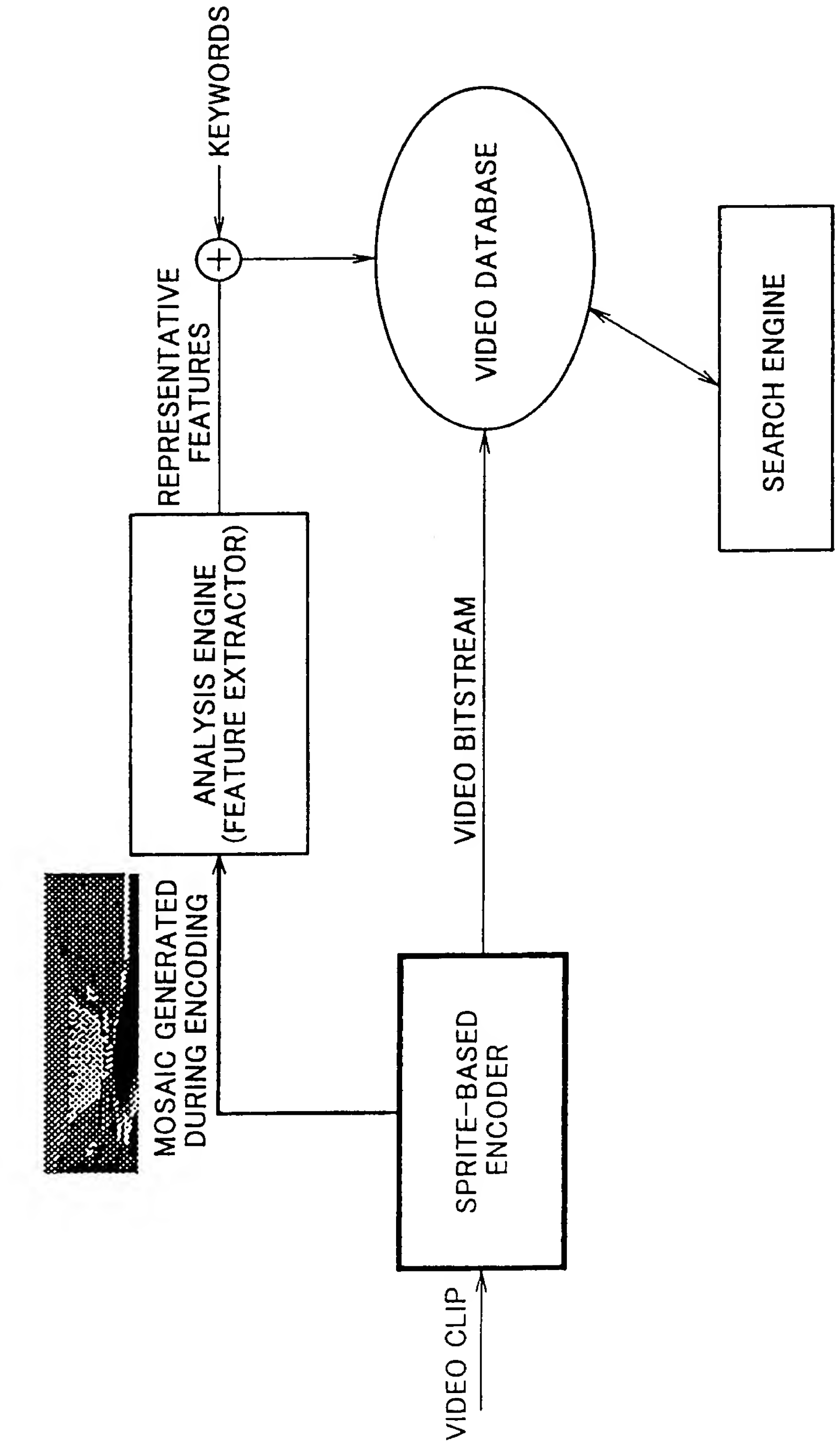

VIDEO CLIP
SPRITE–BASED ENCODER
MOSAIC GENERATED DURING ENCODING
ANALYSIS ENGINE (FEATURE EXTRACTOR)
REPRESENTATIVE FEATURES
KEYWORDS
+
VIDEO BITSTREAM
VIDEO DATABASE
SEARCH ENGINE

FIG.6

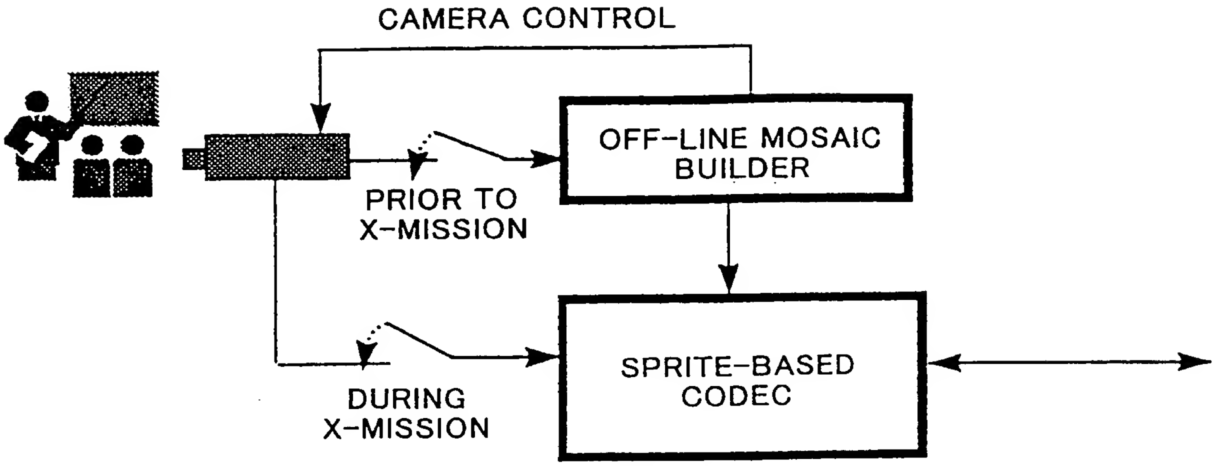
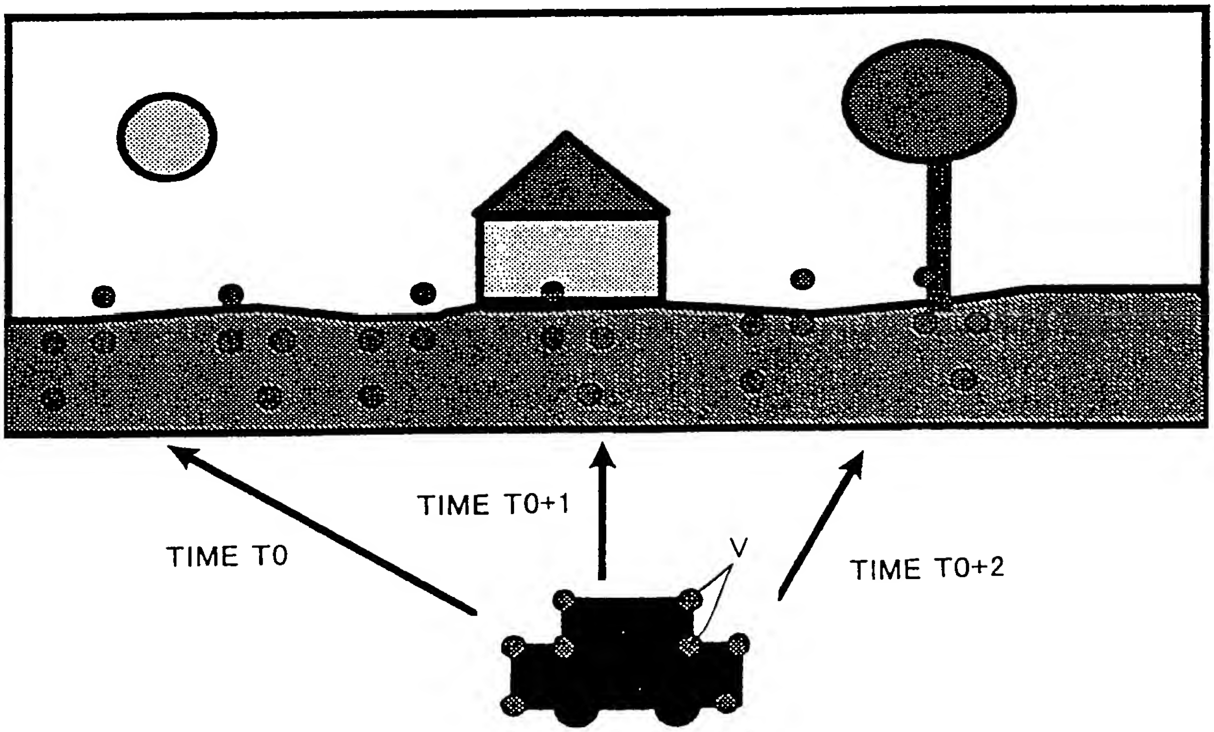

CAMERA CONTROL
OFF-LINE MOSAIC BUILDER
PRIOR TO X-MISSION
DURING X-MISSION
SPRITE-BASED CODEC

FIG.7


TIME T0+1
V
TIME T0
TIME T0+2

INTERNATIONAL SEARCH REPORT

Inter ›nal Application No
PCT/JP 97/04814

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 G06T9/00

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 G06T

Documentation searched other than minimumdocumentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
P,X	US 5 692 063 A (LEE MING-CHIEH ET AL) 25 November 1997 see column 10, line 11 - line 14; figure 26 see column 33, line 41 - column 37, line 55 ---	1,2
A	WANG J Y A ET AL: "REPRESENTING MOVING IMAGES WITH LAYERS" IEEE TRANSACTIONS ON IMAGE PROCESSING, vol. 3, no. 5, 1 September 1994, pages 625-638, XP000476836 --- -/--	

☒ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publicationdate of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

1

Date of the actual completion of theinternational search 6 April 1998	Date of mailing of the international search report 15/04/1998
Name and mailing address of the ISA European Patent Office, P B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax (+31-70) 340-3016	Authorized officer Pierfederici, A

INTERNATIONAL SEARCH REPORT

Inte nal Application No
PCT/JP 97/04814

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication.wnere appropnate, of the relevant passages	Relevant to claim No
A	SZELISKI R: "IMAGE MOSAICING FOR TELE-REALITY APPLICATIONS" PROCEEDINGS OF THE IEEE WORKSHOP ON APPLICATIONS OF COMPUTER VISION, May 1994, pages 44-53, XP002048809 ---	
A	IRANI M ET AL: "Video compression using mosaic representations" SIGNAL PROCESSING. IMAGE COMMUNICATION, vol. 4, no. 7, November 1995, page 529-552 XP004047098 -----	

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/JP 97/04814

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5692063 A	25-11-97	NONE	